移动群智感知中的空间任务分配机制

邢　倩，孙学梅，苑春苗

(天津工业大学 计算机科学与软件学院，天津 300387)

摘　要：在移动群智感知的空间任务分配问题中用户与任务的空间距离直接影响完成任务所需的成本，而现有的研究在这方面却考虑不足，因此以最小化感知成本为目标设计了移动群智感知中的空间任务分配机制。首先，以感知成本最小为目标，基于遗传算法和贪心算法设计了一种高效的任务分配方法；其次，针对用户感知质量的随机性，基于用户的历史感知情况和当前任务的执行情况设计了用户感知质量的更新机制。为验证所提机制的效果，通过仿真实验与两种基准的任务分配方法作比较。实验结果表明，所提机制在感知总成本和用户执行任务所移动的总距离等方面均有更好的效果，因此该空间任务分配机制具有很好的应用前景。

关键词：移动群智感知；空间任务；任务分配；成本最小

中图分类号：TP393.06　　doi: 10.3969/j.issn.1001-3695.2018.09.0644

Assignment mechanism for spatial tasks in mobile crowd sensing

Xing Qian, Sun Xuemei, Yuan Chunmiao

(*School of Computer Science & Software Engineering, Tianjin Polytechnic University, Tianjin* 30000, *China*)

Abstract: For the allocation of spatial tasks in mobile crowd sensing, the spatial distance between users and tasks directly affects the cost required to complete the tasks, while the existing research about this is considered inadequate. In order to minimize cost, the paper proposed a spatial task allocation mechanism. Firstly, based on genetic algorithm and greedy algorithm, it designed an efficient task allocation method to minimize the cost required to complete the tasks. Secondly, considering the randomness of user sensing quality, it designed a mechanism of user sensing quality updating based on the historical quality and the quality of the current task. To verify the effectiveness of the proposed mechanism, it conducted simulations compared with the two benchmarks. The results show that for spatial task allocation the proposed mechanism has better results in terms of total cost and spatial distance that the user needs to move to perform the task. Therefore, the allocation mechanism proposed has better foreground.

Key words: mobile crowd sensing; spatial tasks; task allocation; minimize cost

0　引言

如今，智能手机逐渐成为移动通信设备的主流，这些智能手机通常配有功能强大的传感器[1]，如 GPS，相机和数字式罗盘等。在此背景下，借助智能手机收集感知数据的新兴感知模式—移动群智感知(mobile crowd sensing，MCS)正逐渐成为学术界研究的热点课题[2]。MCS 通常由三部分组成：任务所有者、云端平台和感知用户[3]，其中任务所有者将感知任务提交到云端平台；云端平台通过智能分配算法将感知任务分配给感知用户；感知用户按要求完成感知任务并上传数据。如何选择合适的感知用户是 MCS 的关键问题之一，合理的任务分配方案是感知任务高效执行的前提[4]。

目前，MCS 的应用中大部分都是空间任务，如交通规划[5]、环境监测[6]和公共安全[7]等，这对 MCS 的任务分配方法提出了更高的要求。目前，有关 MCS 中任务分配问题的研究有很多，但是 MCS 的发展处于初期阶段，很多方法和理论仍需完善。文献[8]考虑到感知用户收集数据质量的不确定性，设计出 BLISS 任务分配框架不断重组感知用户集合。文献[9]考虑基于连续时间窗口的任务分配问题，并利用动态规划算法解决该问题。文献[10]将任务分配问题抽象成整数规划问题，然后采用贪心技术实现用户与任务之间的高效匹配。但是上述研究都不适用于空间任务的分配问题。文献[11]将空间任务的分配问题描述成多条件的背包问题，并使用贪心策略解决该问题。文献[12]针对具有时间限制和位置限制的任务提出一种多任务的分配框架，并引入排队机制、信誉机制和委托人机制以得到高效的任务分配方案。在空间任务的分配问题中，用户到任务的空间距离直接影响着感知成本，而文献[11,12]所提出的任务分配方法均是以数据收益为优化目标，并未考虑用户与任务的空间距离对感知成本的影响。文献[13]主要针对多任务分发问题提出了 MultiTasker 方法，目标是使用户完成任务所移动的总距离最短，并且完成任务的用户数最少以优化用户资源，但是该方法没有考虑任务对感知数据质量的要求。

本文在总结现有任务分配方法的基础上，针对空间任务的分配问题，以感知成本最小为优化目标，设计了一种空间任务的分配机制。通过仿真实验将本文的分配算法 GGA-TA 与两种基准的分配方法做比较，并对实验结果进行了分析。

1　问题模型

假设多个任务发布者在云端平台发布了不同地点的 N 个感知任务，用集合 $T=\{1,2...N\}$ 表示。$L_i=\{l_{i,1},l_{i,2}\}$ 表示任务 i 的空间位置，其中 $l_{i,1}$ 和 $l_{i,2}$ 分别表示 2D 空间内一点的横纵坐标。

收稿日期：2018-09-09；**修回日期：**2018-10-24

作者简介：邢倩（1993-），女，硕士研究生，主要研究方向为无线传感器网络、移动群智感知等（13502196367@163.com）；孙学梅，女，硕导，副教授，博士，主要研究方向为无线网络；苑春苗，女，副教授，主要研究方向为计算机网络、网络多媒体.

在 MCS 中，感知任务除了具有感知地点的限制以外，对感知数据的质量也有一定的要求，任务所有者通常要求用户采集到的感知数据质量不低于某临界值[14]。将感知数据质量定义为合格数据的总量，同时满足清晰度要求、感知时间要求和感知地点要求的数据为合格数据。任务 i 的感知数据质量要求用 q^i 表示。假设平台中共有 M 个注册用户，这些用户等待平台为其分配感知任务，用集合 $U=\{1,2...M\}$ 表示用户集合。MCS 中的感知用户不再是专业人员而是普通群众，所以用户感知质量的高低各不相同。定义用户的感知质量为执行一次任务所采集的合格数据的总量，λ_i 表示用户 i 的感知质量；f_i 表示平台需要支付给用户 i 的固定激励，由于用户执行感知任务需要消耗设备电量，承担泄露个人隐私的风险，所以用户 i 每执行一个感知任务就需要平台支付一次固定激励 f_i；p_i 表示用户 i 移动单位距离的成本；受设备能量的限制，用户 i 能够执行的任务个数是有限的[15]，用 n_i 表示用户 i 的最大执行能力，即本次任务分配中最多能执行的任务数。任务发布者发布空间任务后，平台获取任务信息和用户信息并根据智能分配算法合理的分配任务。为降低感知成本，每个任务由一个感知用户执行，但用户的感知质量必须大于或等于任务的感知质量要求以保证任务被成功执行，并且每个用户执行的任务数不得超过该用户的最大执行能力。

在 MCS 中，云端平台往往希望感知成本达到最小，这里所说的感知成本主要包括支付给被选用户的固定激励和动态激励(移动单位距离的动态成本与移动总距离的乘积)。本文目标是站在平台的角度设计一种空间任务的分配机制，同时考虑任务的质量要求、用户的感知质量、最大执行能力和用户与任务的空间距离等因素，力求在满足任务质量要求的同时最小化感知所需的成本。本文目标函数如式(1)所示。

$$\min\ z=\sum_{k=1}^{M}\sum_{i=0}^{N}\sum_{j=0}^{N}p_k X_{i,j,k}d_{i,j}+\sum_{k=1}^{M}\sum_{i=0}^{N}\sum_{j=0}^{N}f_k X_{i,j,k} \quad (1)$$

其中：$X_{i,j,k}$ 为二进制变量，表示用户 k 是否执行空间任务 i 后前往执行任务 j，如果是，$X_{i,j,k}$ 的值为 1，反之为 0。为表述方便，本文假设用户从同一地点出发，完成所分配的任务后返回出发点。加入虚拟任务点 0 作为用户的出发点，设 $T'=\{1,2,\cdots,N\}$。式(1)中第一部分表示支付给被选用户的动态激励。$d_{i,j}$ 表示任务 i 和任务 j 之间的欧几里德距离。式(1)第二部分表示平台支付给用户的固定激励。

目标式(1)需要遵循以下几个限制条件：

$$\lambda_k \ge y_{i,k}q^i \quad u_k\in U \quad t_i\in T \quad (2)$$

$$\sum_{i=1}^{N}y_{i,k}\le n_k \quad u_k\in U \quad (3)$$

$$\sum_{k=1}^{M}y_{i,k}=1 \quad t_i\in T \quad (4)$$

$y_{i,k}$ 为二进制变量，表示用户 k 是否执行了任务 i，如果是，$y_{i,k}$ 的值为 1，反之为 0。约束条件式(2)确保执行每个任务的用户的感知质量都满足该任务的质量要求。约束条件式(3)确保每个用户执行的任务数不超过用户的最大执行能力。约束条件式(4)表示每个任务 i 被执行一次。

2　空间任务分配机制

为解决第二部分所提出的空间任务的分配问题，设计了一种空间任务分配机制，主要内容包括两部分：a)以感知成本最小为目标，基于遗传算法和贪心算法混合的空间任务分配方法(greedy and genetic algorithm for tasks allocation, GGA-TA)；b)用户感知质量的更新机制。

2.1　空间任务分配方法 GGA-TA

移动群智感知中的空间任务分配问题属于典型的 NP 难问题，遗传算法是根据自然界中生物进化规律衍化而来的一种搜索启发式算法，可以用来解决复杂的优化问题。因此使用遗传算法解决上述空间任务分配问题。但是移动群智感知中的任务分配问题规模较大，传统的遗传算法很难得到满意的分配方案，所以设计了遗传算法和贪心算法混合的空间任务分配方法 GGA-TA。

1)染色体编码及适应度函数

GGA-TA 采用自然数编码方式，0,1,2... M 表示 M 个用户，用户所在的位置下标对应任务。如图 1 所示，假设平台中有 8 个任务，3 个用户，图 1 染色体表示用户 1 执行任务 1、4、6；用户 2 执行任务 3、5、8；用户 3 执行任务 2、7。

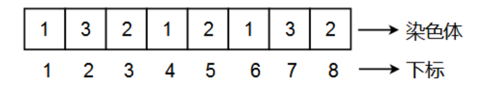


图 1　染色体编码方式

Fig.1 Chromosome coding

上述编码方式产生的分配方案能够保证每个任务被执行一次，但不能保证分配给每个任务的用户的感知质量符合任务的质量要求，也不能保证分配给用户的任务数量在其最大执行能力的范围内。为了处理式(2)(3)所示的约束条件，GGA-TA 方法在计算染色体评价值时引入惩罚系数 p_w，染色体的评价值的具体计算过程如算法 1 所示。为提高算法的效率，GGA-TA 方法在基本遗传算法的基础上加入贪心算法用于规划单个感知用户的路径，用户首先完成距离当前位置最近的任务，如算法 1 中步骤 c）d）所示。

算法 1　计算染色体的评价值

输入：染色体 C，$Q=(q^1,q^2...q^N)$，$\Gamma=(\lambda_1,\lambda_2...\lambda_M)$，$F=(f_1,f_2...f_M)$，$P=(p_1,p_2...p_M)$，$N=(n_1,n_2...n_M)$，$Des=(des_1,des_2...des_M)$。

输出：E。

a)确定染色体 C 中被选用户的集合 $S=\{1,2...s\}$，令 $k=1$、$E=0$。

b)如果 $k>s$ 跳转到步骤 5；否则根据染色体 C 确定用户 k 要执行的任务集合 $R_k=\{1,2...r\}$，令 $E=E+f_k\times|R_k|$，如果 R_k 中元素的个数 $|R_k|$ 大于用户 k 的执行能力 n_k，令 $E=E+p_w$。

c)如果 R_k 为空，令 $E=E+\text{distance}(k,des_k)*p_k$，$k=k+1$ 重复步骤 b)；选择 R_k 中距离用户 k 当前位置最近的任务 j，令 $E=E+\text{distance}(k,j)*p_k$，并将任务 j 的位置作为用户 k 的当前位置。如果任务 j 的质量要求 q^j 大于用户 k 的感知能力 λ_k，令 $E=E+p_w$。

d)从集合 R_k 中移除任务 j，跳转到步骤 c)

e)输出染色体 C 的评价值 E。

算法 1 中 $Des=(des_1,des_2...des_M)$ 表示用户执行任务后返回的地点，其中 $des_k=(des_{k,i},des_{k,j})$。第一节中为表述方便假设所有用户从同一地点出发，执行任务后返回出发点，但从算法 1 中可知，GGA-TA 算法在用户从不同地点出发，执行任务后不返回出点的情景中依然可行。

式(5)为计算染色体的适应度函数。其中 E 表示某一染色体的评价值；$fitness$ 表示该染色体的适应度值；$\max E$ 表示当前种群中的最大评价值；$\min E$ 表示当前种群中的最小评价值。本文中染色体的评价值越小，$fitness$ 越大，染色体的适应度越高。

$$\frac{1}{fitness}=\frac{E-\min E}{\max E-\min E} \tag{5}$$

2）种群初始化

在遗传算法中初始种群极大的影响着算法结果的好坏，移动群智感知中空间任务分配问题的规模较大，因此随机产生初始种群不利于快速收敛到最优结果。GGA-TA 算法中根据任务的质量要求生成初始种群如算法 2 所示。

算法 2　生成初始种群

输入：N，M，$Q=(q^1,q^2...q^N)$，$\Gamma=(\lambda_1,\lambda_2...\lambda_M)$，$Size$。

输出：$IN=\begin{bmatrix} c_{1,1}c_{1,2}\cdots c_{i,N} \\ ... \\ c_{Size,1}c_{Size,2}\cdots c_{Size,N}\end{bmatrix}$。

a)令 $k=1$；

b)如果 $k>Size$，输出 IN；否则令 $i=1$。

c)如果 $i>N$，令 $k=k+1$，跳转到步骤 a)。

d)确定任务 i 的质量要求 q^i，从用户集中查找所有感知质量大于或等于 q^i 的用户子集 $S_i=\{1,2...s\}$，从用户子集中随机选择一个用户标志赋值给 $c_{k,i}$，令 $i=i+1$，跳转到步骤 c)。

3)选择算子

遗传算法中的选择算子用于将适应度高的优秀个体保存到下一代，将适应度低的个体淘汰。GGA-TA 算法的选择算子将种群中的所有的个体按适应度值从大到小降序排列，保留种群中前一半数量的染色体，将后一半数量的染色体淘汰。

4)交叉算子

GGA-TA 算法采用部分匹配交叉，即随机选择两个染色体并随机设置两个交叉点，交换两个染色体所设定的两个交叉点之间的染色体片段。

5)变异算子

变异算子是为了避免算法结果陷入局部最优，GGA-TA 算法的变异算子采用单点变异，即随机产生选择变异点。变异点对应任务 i，i 的质量要求为 q^i，为加快算法的收敛速度，随机选择感知质量大于或等于 q^i 的用户赋值给变异点。

6)终止条件

GGA-TA 算法达到最大进化代数时终止运行。GGA-TA 具体内容如算法 3 所示。

算法 3　GGA-TA 算法

输入：N，M，$Q=(q^1,q^2...q^N)$，$\Gamma=(\lambda_1,\lambda_2...\lambda_M)$，$F=(f_1,f_2\cdots f_M)$，$P=(p_1,p_2\cdots p_M)$，$N=(n_1,n_2\cdots n_M)$，$Size$，$G$，$p_c$，$p_m$。

输出：Θ。

a)令 $t=1$，调用算法 2 生成初始种群 IN_1。

b)令 $t=t+1$，如果 $t>G$，跳转到步骤 6；否则调用算法 1 计算 IN_{t-1} 中所有染色体的评价值，根据式(5)计算 IN_{t-1} 中所有染色体的适应度。

c) IN_{t-1} 中的染色体按照适应度值降序排列，将前 $\left\lceil \frac{Size}{2} \right\rceil$ 个染色体赋值给 IN_t，令 $i=1$。

d)如果 $i>\left\lceil \frac{Size}{2} \right\rceil-1$，令 $j=1$ 跳转到步骤 e）；否则在 IN_t 前 $\left\lceil \frac{Size}{2} \right\rceil$ 个染色体中随机选择两个染色体 In_1 和 In_2，生成[0,1]的随机数 r，如果 $r\geq c$，令 $In_3=In_1, In_2=In_4$。否则随机生成两个交叉点，交换 In_1, In_2 在两个交叉点之间的部分染色体得到两个新的染色体 In_3 和 In_4。

e)将 In_3，In_4 加入到 IN_t 中，令 $i=i+2$，跳转到步骤 d)。

f)如果 $j>Size$，令 $t=t+1$，跳转到步骤 b)；否则生成[0,1]的随机数 r，如果 $r\geq m$，令 $j=j+1$，重复步骤 e)；否则在 IN_t 中随机选择一个染色体 C_l，随机生成一个变异点，变异点位置对应任务 k，从集合 U 中选择感知质量大于或等于 q^k 的用户标志，赋值给 $c_{l,k}$ 令 $j=j+1$。

g)将 IN_{t-1} 中适应度最小的染色体赋值给 Θ。

GGA-TA 算法的选择算子将适应度值高的一半数量的染色体保留如算法 3 中的步骤 c)所示，然后以被保留的一半数量的染色体作为父代生成另一半数量的染色体共同构成一个种群如算法 3 中的步骤 d)e)所示。对选择算子和交叉算子构成种群执行单点变异操作构成新一代种群，如算法 3 中的步骤 f)所示。当 GGA-TA 算法达到终止条件时，输出最后一个种群中适应度最高的染色体作为任务分配方案。表 1 中对常用的参数进行描述。

表 1　常用参数

Table 1　Common parameters

N	任务数量
M	用户数量
Q	$Q=(q^1,q^2...q^N)$，q^i 表示任务 i 的质量要求
Γ	$\Gamma=(\lambda_1,\lambda_2...\lambda_M)$，$\lambda_i$ 表示用户 i 的感知质量
F	$F=(f_1,f_2\cdots f_M)$，f_i 表示用户 i 的固定激励
P	$P=(p_1,p_2\cdots p_M)$，p_i 表示用户 i 的动态成本
E_i	染色体 C_i 的评价值
$Size$	种群规模
G	最大迭代次数
p_w	惩罚系数
p_c	交叉概率
p_m	变异概率
Θ	任务分配方案

2.2　用户感知质量的更新机制

MCS 中感知用户的非专业性造成用户每次完成任务的感知质量是不可预知的随机值，但是可以通过总结用户历史的感知情况得到用户感知质量的估计值。为得到更接近用户的实际情况的估计值，本文加入用户感知质量的更新机制，每执行完一个任务后，根据用户上传的感知数据更新用户的感知质量，为后续的任务分配做准备。用户感知质量的更新公式如式(6)所示。其中 λ_k^r 表示感知用户 u_k 执行完 r 个任务后的感知质量。A_k^j 表示用户 u_k 执行任务 t_j 时上传的合格数据的总量。α 为常量，通过调整 α 的大小，可以调整用户执行当前任务的结果对感知质量更新的影响。为避免用户执行任务时的偶然性，用户感知质量的更新过程中增大历史感知质量的比重，因此设置 α 的取值范围在区间[0.5,1]。

$$\lambda_k^r=\alpha\lambda_k^{r-1}+(1-\alpha)A_k^j \tag{6}$$

3　仿真实验

针对 MCS 空间任务的分配问题，综合考虑了用户到任务的空间距离、任务的质量要求、用户的感知质量及最大执行能力等因素，以感知成本最小为目标，基于遗传算法和贪心算法设计了一种空间任务的分配算法 GGA-TA。刘琰等人[13]针对多任务情景，以感知成本最小为目标设计了一种 PT-Most 算法，但是该算法并未考虑用户的感知质量和任务的质量要求，为了和 PT-Most 算法进行比较证明 GGA-TA 算法的高效性，在 PT-Most 算法第一步中分别优先选择固定激

励低的用户和感知质量与固定激励比值高的用户，并用 CostFirst 和 RatioFirst 分别表示两种算法，为用户分配任务时，设置约束条件：不满足任务质量要求的用户不能执行该任务。

在 MATLAB 实验环境中模拟 N 个感知任务，随机分布在 600×600 （单位:km）的区域内。每个任务的最低质量要求 q^i 服从区间[0,30]上的均匀分布。此外，模拟 M 个感知用户，用户的出发点设为(0,0)，每个用户的感知质量 λ_k 服从区间[0,30]上的均匀分布。用户的固定激励 f_k 服从区间[1,30]上的均匀分布，用户的动态成本 p_k 服从区间[0.1,0.3]上的均匀分布。感知用户的最大执行能力服从区间[4,8]上的均匀分布。其他参数设置如表 2 所示。为保证算法结果近似收敛到最优，仿真实验中设置群体规模($Size$)为 100，最大进化代数(G)为 1500。,为处理约束条件，惩罚系数(p_w)取 100000。交叉概率和变异概率是影响遗传算法性能的关键所在，为避免实验结果陷入局部最优，本次仿真实验设置交叉概率(p_c)为 0.9，变异概率(p_m)为 0.05。每次仿真实验 GGA-TA 算法执行 10 次，分别取 10 次实验结果的最小值和平均值。

表 2　参数设置表

Table 2　Parameter settings

参数名称	值
$Size$	100
G	1500
p_w	1000000
p_c	0.9
p_m	0.05

3.1　用户总量变化对三种算法性能的影响

设置任务数量 N =10 和 N =20，分别观察任务数量较少和任务数量较多的情况下，用户总量变化对三种算法性能的影响。

a)任务数量 N =10,结果如图 2、3 所示。从图 2 和 3 中可以看出任务数量为 10 时，随着用户总量的增加，GGA-TA 算法的感知总成本和用户移动的距离先下降后基本保持不变，而 CostFirst 和 RatioFirst 算法的感知成本和用户的移动的距离远远高于 GGA-TA 算法,这是因为 GGA-TA 是一种智能算法，能够综合考虑任务的质量要求，用户的感知质量，用户与任务的空间距离等多个因素找到最优解，而 CostFirst 和 RatioFirst 仅仅使用贪心算法容易陷入局部最优。

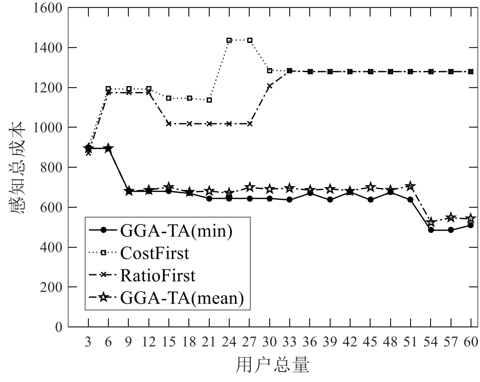


图 2　N=10 时，用户总量与感知总成本的关系

Fig. 2　Effect of number of users on the cost of three algorithms when there have ten tasks.

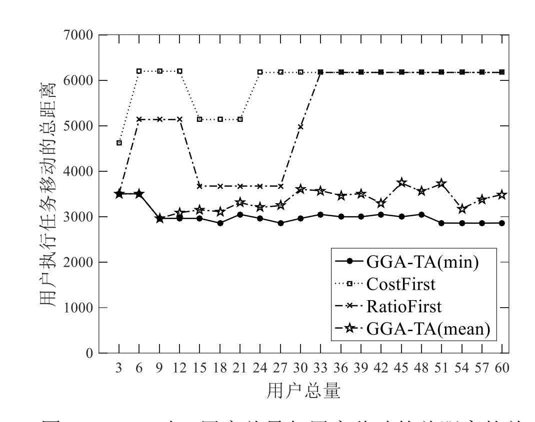


图 3　N=10 时，用户总量与用户移动的总距离的关系

Fig. 3　Effect of number of users on the distance of three algorithms when there have ten tasks.

b)任务数量 N =20,结果如图 4、5 所示。

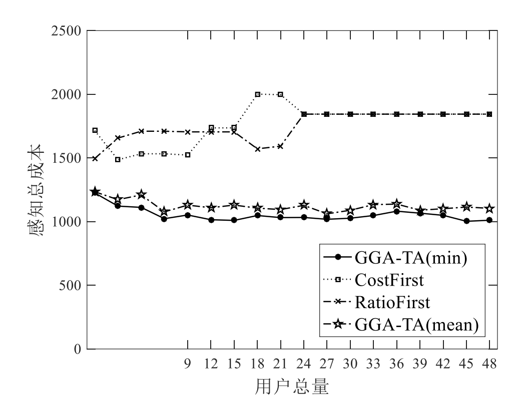


图 4　N=20 时，用户总量与感知总成本的关系

Fig. 4　Effect of number of users on the cost of three algorithms when there have twenty tasks.

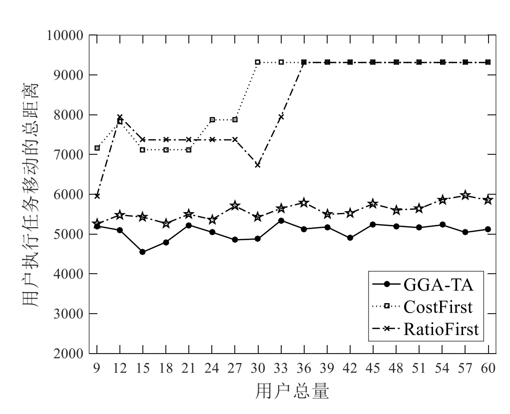


图 5　N=20 时，用户总量与用户移动的总距离的关系

Fig. 5　Effect of the number of users on the distance of three algorithms when there have twenty tasks.

从图 4、5 中可以看出任务数量为 20 时，随着用户总量的增加，GGA-TA 算法感知总成本和用户移动的距离基本保持不变而且远远低于 CostFirst 和 RatioFirst 算法。

3.2　任务总量变化对三种算法性能的影响

为分析任务总量变化对三种方法性能的影响，实验设置用户总量 M =20，结果如图 6、7 所示。

由图 6、7 可知，用户总量 M =20 时，随着任务总量的增加，三种算法的感知总成本和用户移动的距离均不断增加。但是当任务总量达到 36 时，RatioFirst 算法无法将所有任务分配出去，当任务总量达到 51 时，CostFirst 算法无法将所有

任务分配出去。当任务总量超过 48 时，GGA-TA 算法的一次计算结果存在不能分配失败的情况，但 10 次计算后依然能将任务成功分配。而且 GGA-TA 算法在感知总成本和用户移动的总距离方面总是小于前面两种算法。

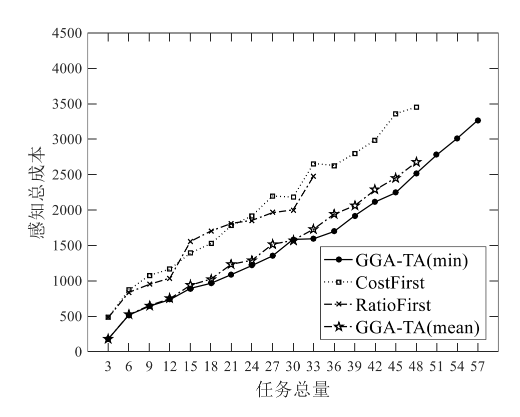


图 6　M=20 时，任务总量与感知总成本的关系

Fig. 6　Effect of number of tasks on the cost of three algorithms when there have twenty users.

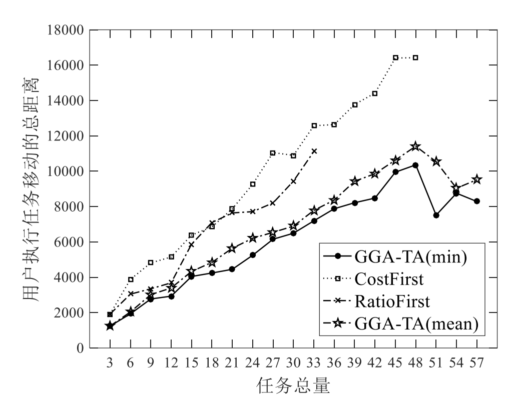


图 7　M=20 时，任务总量与用户移动的总距离的关系

Fig. 7　Effect of number of tasks on the distance of three algorithms when there have twenty users.

为分析任务总量变化对三种算法性能的影响，记录了用户总量 M =20 任务总量不断增加的情况下三种算法的运行时间，如图 8 所示。

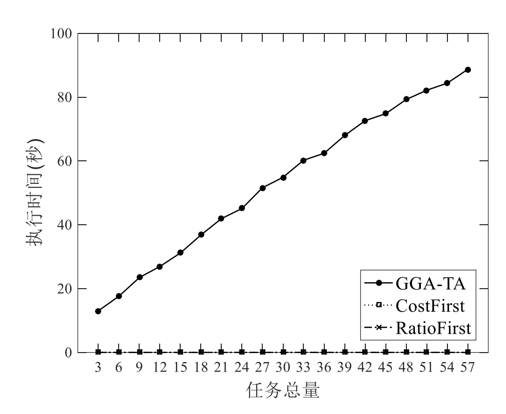


图 8　M=20 时，任务总量与算法执行时间的关系

Fig. 8　Effect of the number of tasks on the execution time of three algorithms when there have twenty users.

如图 8 可知，随着任务总量的增加，GGA-TA 的计算量不断增大，算法执行时间也不断增大。由于 GGA-TA 算法为启发式算法，需要经过不断的迭代计算才能得到最优结果，所以 GGA-TA 算法的执行时间远高于另外两种算法。

实验证明，相比 CostFirst 算法和 RatioFirst 算法，本文基于遗传算法和贪心算法混合提出的 GGA-TA 算法，能在任务全部被成功分配的前提下实现感知成本最小。在本文所阐述的系统模型中，GGA-TA 算法能够综合考虑任务的质量要求、用户的感知质量、用户的最大执行能力和用户到任务的空间距离等多种因素，使得该方法相比另外两种算法在用户执行任务移动的总距离和感知总成本等方面均有较好的表现。

4　结束语

针对 MCS 中的空间任务分配问题，本文站在平台的角度，设计了一种空间任务的分配机制，旨在满足任务质量要求的同时最小化感知成本。实验证明，本文提出的分配机制在小规模的任务分配问题中具有良好的效果。但是 GGA-TA 算法的执行时间随着任务量的增大不断增加，所以本文所提机制不适用于大规模计算问题。在以后的研究中力求设计一种适应于大规模空间任务分配问题的算法。

参考文献：

[1] Yin Xizhe, Shen Weiming, Samarabandu J, *et al.* Human activity detection based on multiple smart phone sensors and machine learning algorithms [C]//Proc of the 19th IEEE International Conference on Computer Supported Cooperative Work in Design. Piscataway,NJ: IEEE, 2015: 582-587.

[2] Ganti R K, Ye Fan, Lei Hui. Mobile crowdsensing: current state and future challenges [J]. IEEE Communications Magazine, 2011, 49(11): 32-39.

[3] 吴垚，曾菊儒，彭辉，等. 群智感知激励机制研究综述 [J]. 软件学报，2016, 27(8): 2025-2047. (Wu Yao, Zeng Juru, Peng Hui, *et al.* Survey on Incentive Mechanisms for Crowd Sensing [J]. Journal of Software, 2016, 27(8): 2025-2047)

[4] Karaliopoulos M, Telelis O, Koutsopoulos I. User recruitment for mobile crowdsensing over opportunistic networks [C]//Proc of IEEE Conference on Computer Communications.Piscataway,NJ: IEEE, 2015: 2254-2262.

[5] Coric V, Gruteser M. Crowdsensing maps of on-street parking spaces [C]//Proc of IEEE International Conference on Distributed Computing in Sensor Systems. Piscataway,NJ:　IEEE, 2013: 115-122.

[6] Maisonneuve N, Stevens M, Niessen M, *et al.* Noisetube: measuring and mapping noise pollution with mobile phones [C]// Proc of the 4th International Symposium on Information Technologies in Environmental Engineering. Piscataway,NJ: IEEE, 2009: 215-228.

[7] Guo Bin, Chen Huihui, Yu Zhiwen, *et al.* FlierMeet: a mobile crowdsensing system for cross-space public information reposting, tagging, and sharing [J]. IEEE Trans on Mobile Computing, 2015, 14 (10): 2020-2033.

[8] Han Kai, Zhang Chi, Luo Jun. Taming the uncertainty: budget limited robust crowdsensing through online learning [J]. IEEE/ACM Trans on Networking, 2016, 24(3): 1462-1475.

[9] Xu Jia, Xiang Jinxin, Yang Dejun. Incentive mechanisms for time window dependent tasks in mobile crowdsensing [J]. IEEE Trans on Wireless Communications, 2015,11(14): 6353-6364.

[10] 施战，辛煜，孙玉娥，等. 基于用户可靠性的众包系统任务分配机制 [J]. 计算机应用, 2017, 37(9): 2449-2453. (Shi Zhan, Xin Yu, Sun Yu-e, *et al.* Task allocation mechanism for crowdsourcing system based on reliability of users [J]. Journal of Computer Applications, 2017, 37(9):

2449-2453)

[11] Miao Chunyan, Yu Han, Shen Zhiqi, *et al.* Balancing quality and budget considerations in mobile crowdsourcing [J]. Decision Support Systems, 2016(90): 56-64.

[12] Estrada R, Mizouni R, Otrok H, *et al.* A crowd-sensing framework for allocation of time-constrained and location-based tasks [J]. IEEE Trans on Services Computing, 2018, DOI: 10. 1109//TSC. 2017. 2725835.

[13] 刘琰，郭斌，吴文乐，等. 移动群智感知多任务参与者优选方法研究 [J]. 计算机学报，2016,40(8): 1872-1877. (Liu Yan, Guo Bin, Wu Wenle, *et al.* Multitask-Oriented Participant Selection in Mobile Crowd Sensing [J]. Chinese Journal of Computers, 2016,40(8): 1872-1877)

[14] Wang Jing, Tang Jian, Xue Guoliang, *et al*. Towards energy-efficient task scheduling on smartphones in mobile crowd sensing systems [J]. Computer Networks, 2017 (115): 100-109.

[15] Liu C H, Zhang B, Su X, *et al.* Energy-aware participant selection for smartphone-enabled mobile crowd sensing [J]. IEEE Systems Journal, 2017, 11(3): 1435-1446.